DOCUMENT RESUME

ED 274 340 IR 012 332

AUTHOR	Morris, Charles L., Jr.; And Others
TITLE	System Design Considerations for Microcomputer Based Instructional Laboratories. Final Report.
INSTITUTION	Naval Training Analysis and Evaluation Group, Orlando, Fla.
REPORT NO	SR-86-004; TR-86-006
PUB DATE	Apr 86
NOTE	52p.
PUB TYPE	Guides - Non-Classroom Use (055) -- Reports - Research/Technical (143)
EDRS PRICE	MF01/PC03 Plus Postage.
DESCRIPTORS	*Authoring Aids (Programing); *Computer Assisted Instruction; Computer Graphics; *Computer Managed Instruction; Computer Simulation; *Computer Software; Guidelines; Input Output Devices; Instructional Systems; *Microcomputers; Military Training; Programing Languages; *Systems Development; Technical Education
IDENTIFIERS	Navy

ABSTRACT

This report provides detailed guidelines for configuring a computer based instructional (CBI) laboratory based on the use of Navy/Air Force standard microcomputer hardware and software, some off-the-shelf products, and specialized simulation hardware/software, Navy developed authoring and operating systems, and operational equipment interfaces. The first of four main sections in this report suggests a functional categorization system for CBI and describes the microcomputer hardware and software subsystems needed to satisfy a CBI laboratory requirement. Operating systems, languages, and network options are addressed as component subsystems of a CBI laboratory system. The second section addresses the ways in which a CBI laboratory can be used to support technical training functions including prerequisite skills remediation, independent study, job practice, testing, and refresher training. The third section discusses authoring languages and systems requirements for effective laboratory application and proposes government sponsored development in this area. The last section discusses CBI laboratory system configurations and specifically addresses such technical issues as graphics and simulation that must be considered when selecting a configuration. Several figures are included for illustrative purposes. (DJR)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED274340

TECHNICAL REPORT 86-006

SYSTEM DESIGN CONSIDERATIONS FOR MICROCOMPUTER BASED INSTRUCTIONAL LABORATORIES

APRIL 1986

Charles L. Morris, Jr.
Richard T. Braby
Wayne A. Knight

TRAINING ANALYSIS AND EVALUATION DEPARTMENT
NAVAL TRAINING SYSTEMS CENTER
ORLANDO,FLORIDA 32813-7100

BEST COPY AVAILABLE

IR012332

T.W. McNaney, Director
Training Analysis and
Evaluation Department

W.H. Lindahl, Technical Director
Naval Training Systems Center

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION IS UNLIMITED

GOVERNMENT RIGHTS IN DATA STATEMENT

Reproduction of this publication in whole or in part is permitted for any purpose of the United States Government.

REPORT DOCUMENTATION PAGE

Unclassified

1a. REPORT SECURITY CLASSIFICATION

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited.

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
Special Report 86-004

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
Naval Training Systems Center
Orlando, FL 32813-7100

6b. OFFICE SYMBOL (If applicable)
NAVTRASYSCEN

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS (City, State, and ZIP Code)
Naval Training Systems Center (Code 123)
Department of the Navy
Orlando, FL 32813-7100

7b. ADDRESS (City, State, and ZIP Code)

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL (If applicable)
NAVTRASYSCEN

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS (City, State, and ZIP Code)

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
SYSTEM DESIGN CONSIDERATIONS FOR MICROCOMPUTER BASED INSTRUCTIONAL LABORATORIES

12. PERSONAL AUTHOR(S)
Charles L. Morris, Jr., Richard T. Braby, Wayne A. Knight

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM _____ TO _____	April 1986	53

16. SUPPLEMENTARY NOTATION

17 COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Computer Based Instruction, Training Laboratories, Computer Based Authoring, Two Dimensional (2D) Simulation, Three Dimensional (3D) Simulation.
05	09		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

System design considerations for a computer based instructional (CBI) laboratory are explored and the concept of laboratory hardware and software modularity is presented. The adaptive features of this type of instructional system are discussed and categories of training are addressed in terms of instructional goals and methods. A number of computer hardware and software topics; such as, operating systems, languages, communications, input and output capabilities, and graphics are presented in terms of training related applications. Laboratory configurations are also proposed for typical training applications and the topics of two dimensional (2D) and three dimensional (3D) simulation are addressed. Both 2D and 3D simulation are suggested as preferred instructional design approaches for courseware authoring in the CBI laboratory.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Charles L. Morris, Jr.

22b. TELEPHONE (Include Area Code)
(305) 646-5349/4550

22c. OFFICE SYMBOL
NAVTRASYSCEN Code 123

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete.

EXECUTIVE SUMMARY

Simulation based training devices have been used extensively as a primary training medium for many years in both the public and private sectors. Two of the more visible and memorable applications are the Link "blue box" flight trainers used during World War II to train pilots and the spacecraft trainers used to train astronauts for the Lunar Landing Mission. Throughout this same period of time there were hundreds of less visible major training device applications. These devices include flight trainers for most of the major airlines; surface, subsurface, ground and air trainers for the military services; and industrial trainers such as those used for nuclear power plant training. All of these trainers have utilized relatively complex, high cost computers and simulation software. It is envisioned that this type of training will continue, and that training effectiveness and efficiency improvements will be realized with each new training device generation.

The advances being made today in computer based technology exceed the most optimistic projections of a decade ago. Microcomputers are now available for less than $5,000 which have greater computational power than yesterday's mainframes. In addition, the computer literacy level in the United States is very high in terms of numbers of programmers and their capabilities. Because of this changing technology environment, it is now possible to extend simulation based training capabilities to the broad Navy technical training community. Many Navy maintenance and operator training programs do not utilize simulation based training because, until now, this mode of instruction has been cost prohibitive. The cost of simulation is falling and it is now possible to apply simulation based training capabilities, currently limited to critical skill ratings such as Electronic Warfare, Cryptology, and Sonar, to the broad spectrum of Navy technical training.

To implement this new simulation based training approach in the Navy technical training community, there must be a benefit to offset the costs. It is likely that economic factors will serve as the primary motivating stimulus for development programs of this type. Economic justification should not be difficult for many applications. For economic justification, the broad application of this technology would be presented in measures such as scale economies, new technology applications, lower factor prices, and operations efficiency. For computer based training applications, it is also suggested that the improvement measures will include Government sponsored microcomputer operating system and authoring system development for simulation and training applications. In addition, hardware cost savings could also be realized by utilizing standardized PC board form factors and bus systems for interface and peripheral components.

By providing a path for this new computer based training technology to be applied broadly in the Navy technical training community and by stimulating competition through Government sponsorship of high investment development, Navy manpower and training benefits beyond any considered to date appear possible. It seems reasonable to estimate technical training efficiency improvements of 10 percent or more for both shore based and afloat technical training. If this were to be achieved, it would translate into a savings or added training value of $100 million to $500 million per year.

CONTENTS

Page

INTRODUCTION..... 9

Background..... 10
Purpose..... 11
Organization of Report..... 11

MODULARIZATION OF TRAINING FUNCTIONS HARDWARE AND SOFTWARE..... 13

Instructional Functions..... 13
Microcomputer Hardware..... 15
Microcomputer Software..... 18

CBI LABORATORY INSTRUCTIONAL GOALS AND METHODS..... 21

Instructional Goals..... 21
Instructional Methods..... 27

AUTHORING LANGUAGES AND AUTHORING SYSTEMS..... 35

Authoring Languages..... 35
Authoring Systems..... 37

CBI LABORATORY SYSTEM CONFIGURATIONS..... 41

Simulation..... 41
Graphics..... 43
Laboratory Configurations..... 45

LIST OF ILLUSTRATIONS

Figure Page

1 Training System Functions..... 14

2 Instructional Laboratory Configuration No. 1.. 46

3 Instructional Laboratory Configuration No. 2.. 48

4 Instructional Laboratory Configuration No. 3.. 49

LIST OF TABLES

Table — Page

1 Goals that can be Accomplished in CBI Laboratories..... 25

2 Basic Instructional Methods Used in CBI Laboratories..... 33

INTRODUCTION

Computer based instruction (CBI) has been applied widely during the past decade and a number of applications are considered very successful. However, high confidence training effectiveness data for this relatively new mode of instruction is not yet available. Consequently, there are still many unanswered questions relating to its suitability as a cost-effective training medium. Instructional applications such as drill and practice and testing appear to be very successful when properly implemented. However, concept learning and tutorial applications are not yet widely accepted as an effective alternative to traditional methods. Today, computer simulation is considered to be the most promising instructional technique. New applications continue to evolve for this new technology and with each step forward, a greater degree of acceptance is being noted. Many instructional system designs now include a partial use of computer based training for specific training functions which have been proven effective in other similar applications. Although the computer based segments might only represent 10 to 20 percent of the total training requirement for a specific application, the success being realized is providing a basis for accepting the CBI concept as a viable alternative. The CBI approach discussed in this report is based upon building from the simplest technique to the more complex. Simulation provides a thread through all applications from tutorial to practice and performance testing.

The CBI laboratory is intended to augment traditional instruction as opposed to replacing it. This augmentation is proposed in the form of a generic laboratory configuration with an instructional management station and a number of student stations. It is proposed that instructional augmentation of this type, supported by courseware development tools, will satisfy testing, remediation, refresher training, skill development, and other specific training subfunctions within the total training process. It is envisioned that a CBI laboratory would support a number of courses in a school. One or two hours of testing time a week would satisfy most testing requirements and 8 to 10 hours of laboratory exercise time a week would enhance the training in the laboratory and improve the efficiency of performance measurement. By implementing a CBI instructional laboratory, applications which prove to be cost-effective can be expanded and applications which do not add to training effectiveness or efficiency can be eliminated. This will provide, in effect, an adaptive, client driven, growth capability.

The CBI applications discussed in the body of this report include prerequisite skills remediation, independent study, job practice, testing and refresher training. Within this grouping of training goals, instructional methods are also discussed. A wide range of computer capabilities such as interactive dialogue, graphic presentation, problem generation, response analysis,

remedial feedback and simulation are applied to implement the training sequences selected. It is also possible to provide reports of student progress for all modes of instruction for evaluation by the instructor. Authoring is addressed as a main topic in this report because a CBI system cannot be effectively fielded without a capability for authoring new instructional segments and/or modifying existing segments. In addition, an attempt is made throughout the report to exploit low cost, standardized hardware and software.

RADM R. C. Austin, Chief of Naval Technical Training, as Military Keynote Speaker at the 7th Interservice/Industry Training Equipment Conference, stated that there are approximately 60,000 Navy personnel in school on any given day. If, as proposed in this report, computer based training is to be used to support a maintenance and operator training instructional load of this magnitude, then cost, effectiveness, and standardization considerations are of the utmost importance. These three considerations have been of primary concern during the development of this report.

BACKGROUND

During 1983 the Chief of Naval Education and Training (CNET) tasked the Training Analysis and Evaluation Group (TAEG) to conduct a study to determine the system components required for a range of CAI laboratory configurations. At that time CNET was considering the implementation of Computer Aided Instruction (CAI) at one or more of its school locations and was assessing alternative configurations for development. A study was initiated in response to that tasking and a number of hardware and software options for laboratory components were identified. During the early phases of the study an interim brief of findings was presented to CNET staff members. At that time it was envisioned that two general approaches for laboratory development were feasible. The first approach would utilize systems such as TICCIT, PLATO or WICAT at an average cost of $15,000 to $20,000 per student station. The second approach was based on the use of low cost microcomputers such as Apples, Ataris or Commodores, integrated in the form of a classroom/laboratory network with a central instructor management station and a number of student stations. The cost estimate for the second configuration was approximately $3,000 per student station plus $10,000 to $15,000 for network components and the laboratory management station. Courseware development costs were estimated separately and were in addition to the laboratory implementation cost for both approaches.

Because of funding limitations for new technology applications during this period and also because of the disestablishment of the CNET Staff CAI organization, the study was temporarily suspended. During 1984, the CNET Research Applications Center requested that work continue on the low cost

option for computer based laboratory configurations. It was also at this time that TAEG was organizationally transferred to the Naval Training Equipment Center. The effort continued as a low priority project with the understanding that the study would become a Naval Training System Center (NAVTRASYSCEN) study which would benefit both CNET and the NAVTRASYSCEN.

During this period, there was an interest in broadly applying microcomputers throughout DOD. The Navy and the Air Force were in the process of procuring standard configuration microcomputers for use in both services. This resulted in the first large scale Navy/Air Force microcomputer procurement. Zenith Data Systems won the competition under this procurement strategy and a de facto microcomputer standard for DOD emerged. Zenith used an IBM PC architecture, with some variation, as required by contract and also used the Microsoft Disk Operating System (MS-DOS). This Navy/Air Force standard has evolved to include IBM compatible architectures from other manufacturers, based on the 8088 or 8086 microprocessor and the MS-DOS operating system. As a result of this de facto microcomputer standardization policy in the Navy, the CBI laboratory project was redirected to incorporate this class of microcomputer. The laboratory configurations, discussed in this report, incorporate standard microcomputers and, to the extent possible, standard software. Although this standardization approach can limit the number of viable options for training, it provides a path for large scale application at relatively low cost. This satisfies two of the three objectives stated for the study: standardization and low cost.

PURPOSE

The purpose of this study is to provide a set of guidelines for configuring a CBI laboratory. The configurations proposed are based on the use of Navy/Air Force standard microcomputer hardware and software to the extent possible. Where deviations from standard are necessary to satisfy CBI laboratory training requirements, other off-the-shelf products are recommended. For the remaining cases (standard or off-the-shelf components are not suitable) a description of the product or service is included. Examples of this latter category would be specialized simulation hardware and software, Navy developed authoring systems and operating systems and operational equipment interfaces.

ORGANIZATION OF THE REPORT

In addition to this Introduction, the report is divided into four additional sections. The first section suggests a functional categorization system for CBI and describes the microcomputer hardware and software subsystems needed to satisfy a CBI laboratory requirement. Operating systems, languages, and network options are addressed as component subsystems of the CBI laboratory system. The following section addresses the ways in

which a CBI laboratory can be used to support Navy technical training. The training functions discussed include prerequisite skills remediation, independent study, job practice, testing and refresher training. The next section addresses authoring system requirements for effective laboratory application. This section discusses authoring languages and systems, and proposes government sponsored development in this area. The last section discusses CBI laboratory system configurations and specifically addresses some of the technical issues; such as, graphics and simulation, to be considered when selecting a configuration.

MODULARIZATION OF TRAINING FUNCTIONS HARDWARE AND SOFTWARE

INSTRUCTIONAL FUNCTIONS

The CBI laboratory configurations discussed in this report utilizes computer support for four training system functions as shown in figure 1. The administration function provides support for student record keeping, laboratory resource management, scheduling, and management reporting. The instructional management function controls student assignment and sequencing and includes entry level testing and evaluation, training segment assignment, progress testing, student tracking, and comprehensive performance measurement. The instructional delivery function provides a capability for selecting an instructional strategy and delivery format to facilitate learning and also to provide the instructional sequences to satisfy the training objectives. The authoring function, although not directly integrated with the other functions provides a capability for developing the student materials and computer software for all elements of the laboratory training system.

The administration, management, and delivery functions are interrelated in the sense that delivery is supported by testing and remediation and both are under the control of administrative scheduling.

With an interrelationship such as this, student assignment and tracking can be effectively accomplished while allowing an instructional pace which is consistent with both student ability and instructional objectives. By integrating instructional functions in this way, a complete record of accomplishment is maintained for all types of laboratory instruction. Although the instructional functions are under the control of a single laboratory management computer, they are separated in the form of hardware and software modularization. Each module is independently operable and hardware/software interfaces are utilized to communicate training related data between modules. Modularization of software and hardware is an essential element of design for an effective implementation of the laboratory concept. As an example, the new student assignment module within the instructional management function must utilize information from student records and entry level testing prior to entering the instructional delivery process. However, drill and practice or refresher training might require only the instructional delivery function.

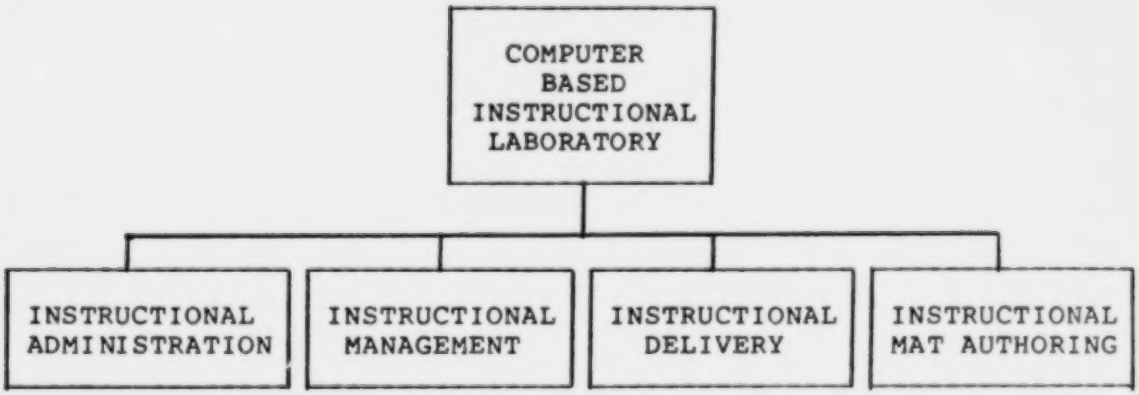


Figure 1. Training system functions

By utilizing an instructional laboratory system design such as this, the assignment, delivery, measurement, and analysis of instruction can be utilized to the extent deemed appropriate for the type of training selected.

MICROCOMPUTER HARDWARE

The microcomputer architecture recommended for the CBI laboratory is based on the IBM PC/XT/AT/RT designs which, at this time, are leading to industry and DOD de facto standards. Current Navy/Air Force microcomputer procurements specify products with these general architectures and most vendors supply products based on these designs. The microcomputer products being procured, typically include an 8088, 8086, or 80286 microprocessor chip, 256K to 640K of random access memory, with expansion capabilities, standard parallel and serial ports and a monochrome or color monitor. In most cases the micocomputers also include a number of mother board slots for input/output bus access. The bus system allows for the use of a variety of input and output devices. Although not widely used in Navy training, the bus access system, supported by appropriate interface cards, could be used to sense and control many types of two dimensional (2D) and three dimensional (3D) input and/or output devices for simulation and training applications.

The secondary storage typically used with these microcomputers consists of one or two double sided 360K byte or 1.2M byte floppy disks devices and/or a 10 to 80M byte hard disk drive. Although this capability can be extended, this type of extension is not addressed for the laboratory configurations described in this report.

Input Devices

The standard input devices used with microcomputers are keyboards, joysticks, digitizer pads, touch screens, mouse devices, track balls and light pens. In addition, a number of other input devices, such as bar code readers, electromagnetic radiation sensors, and sound sensors are used for specialized applications. All of these input devices use some form of transducer to convert the sensor signal to an analog or digital voltage which is ultimately converted to a digital form suitable for computer input. Some input devices, such as keyboards and bar code readers generate a digital code which is communicated to the computer for interpretation. Others generate analog voltages which represent physical spatial or environmental states. Two typical sensor applications are electromagnetic radiation sensors used for garage door openers and thermostatic sensors used for temperature control. The analog or digital voltage representations derived from sensor signals such as these could be converted to a digital form for use in a microcomputer based control system. Similar capabilities offer almost unlimited possibilities for training applications.

An interesting point to be considered in terms of training applications is that the computer does not differentiate between an input from a device 50 feet away or one 50 miles away. An input is simply a digital byte or word to be used as an operand. The design engineering capability for PC/XT/AT interfacing is currently available as it has been for many years. However, it is only recently that costs have fallen to a level which makes special purpose interfacing cost justifiable for technical training. Because training is often related to sensing and control mechanisms, the inherent CBI capability to support these features should not be overlooked in the design of a training laboratory. It should be remembered, however, that there is a decided cost advantage in using standard devices such as keyboards for input and two dimensional displays for output. Because of the generic nature of standard input and output devices there is also an advantage gained through the use of standardized software. When a specialized device is used for input or output, the generic standard advantage is not realized. However, there are many training applications, best represented by simulation based training equipment, where the training benefit can offset the higher cost associated with specialized design. The CBI laboratory can be designed to provide applications specific capabilities for any type of instruction, but it is recommended that a design emphasis on generic applications be maintained.

Output Devices

Output devices are configured similarly to input devices with the exception that the digital signals originate in the computer, are routed through a device controller and signal conditioner and typically terminate at a printer, CRT display, motor controller, meter, digital readout or annunciator panel. As with input devices, the list of output devices to be complete, would have to include all electrically actuated mechanisms. This list would be extremely large and in terms of training potential, not very useful. However, it expands the concept of utilizing the training laboratory microcomputer for many applications beyond those traditionally considered. The primary output devices for training laboratory application will be the CRT display and the printer. These output devices are considered the basic minimum.

Communications

Data communications take many forms and each is dependent on the application requirement. Data are communicated from a computer to a CRT monitor for display, to a printer for hard copy, to a magnetic medium for data storage and to a modem for telephone network transmission. It is not difficult to communicate data. What is difficult is the design, development, and implementation of the communication system. The transmitting

node and the receiving node must utilize common protocols and message formats which can be recognized and decoded at both ends of the communication link. For data networks, the data must also adhere to network standards for packetizing and routing. If error detection and correction are required or, if encrypting is to be utilized, another level of complexity is added. However, most of the network complexities are reduced for small networks suitable for classroom, laboratory, or school use.

The network arrangement proposed for a CBI laboratory is of the star configuration variety. The instructor/management station would be the hub and the student stations would communicate via spokes to the hub. For a configuration of this type a relatively inexpensive multiplexed low speed RS-232 link between student stations and management station would suffice. For the typical case, data rates between the instructor station and any student station would be limited to 19,600 baud which is approximately 2000 bytes or characters per second. This is suitable for assignment screens, student progress information and testing but it is not suitable for graphics, simulation, and courseware communication. Data of this type would have to be conveyed by floppy disks distributed by the instructor. Although limited in terms of high speed data interchange, a configuration of this type is suitable for many low cost laboratory applications.

Another communications option is based upon the use of a local area network (LAN). A network of this type uses a bus configuration which is nothing more than a single coaxial or twisted wire cable, running to all stations with taps into the bus at each station. Student stations would contain a network interface card and software. The instructor/management station would contain an interface card and a network cluster controller. With this configuration, any station can communicate with any other station, if programmed for this capability. However, for the CBI laboratory, communication need only be supported between instructor station and each student station. This network arrangement is not significantly different from the RS-232 network configuration except for one very important characteristic. The data communication rate is significantly higher. LANs are available today with data communication rates between 1 Mbit and 10 Mbit per second. At this rate, it is possible to communicate graphics, simulation, and courseware data between the instructor/management station and student stations without any appreciable delays in instructional delivery. Consequently, it is not necessary to use floppy disks at the student station. This allows for lower cost student stations and eliminates the complexity of handling hundreds of floppy disks. The network costs are higher for a configuration of this type and the instructor management station must have a relatively powerful processor with a hard disk storage capability of at least 10 to 80 megabytes. Although the costs are higher, the features

offered make this option very desirable. It will most likely be the preferred option for laboratories with intensive graphics and simulation instructional sequences.

MICROCOMPUTER SOFTWARE

Microcomputer software in concept is not difficult. However, the uniqueness of training applications software introduces some software complexities. Operating systems, authoring languages, applications programs, and utility programs are all computer software. They all reduce to machine language operation codes which can be interpreted and executed by the microprocessor, one instruction at a time. If each type of software was analyzed in terms of processing activity, it would be difficult to differentiate between an instruction for a high level language compiler and an instruction for an applications program. The partitioning of software by function and application is only in the conceptual framework and in the functional modularization of the code. Operating systems, languages and applications programs all use sequences of machine code to accomplish a function. It would not be difficult to produce software products for training applications without a standard operating system. However, development efficiency would be reduced and the application programs would have to include an executive module and an input/output module. A number of Navy training systems have been developed in this way and many are more efficient, in terms of computer resource use, than those developed with standard operating systems and standard languages. This degree of customization, however, carries with it some significant penalties. The software is often difficult to maintain by anyone other than the developer and the programs are not transportable to other computers for other applications. In addition, proprietary software is difficult to differentiate from other applications software.

For a CBI laboratory application, there is currently good reason to follow an approach which uses standard hardware, operating systems and languages. However, for cases wherein training requirements cannot be satisfied because of standard software package limitations, government developed operating systems and languages should be considered.

Operating Systems

A microcomputer system consists of many interrelated subsystems. These subsystems typically include an arithmetic and logic unit (ALU), random access memory (RAM), read only memory (ROM), input channels, output channels, memory management systems with direct memory access (DMA) capabilities, priority interrupt logic and a system clock. The integration of all these computer resources into an operable computer system with program and file handling capabilities, requires a hardware or software executive

controller. This executive system has become known as an operating system. On early computer systems many of these functions were implemented in hardware logic and the programmer had little more than a bootstrap loader to enter programs into the computer. Today, most of the operating system functions are implemented in software with the exception of the input/output control system. The input/output control system is typically implemented with hardwired logic and ROM based machine code which is generally unique to a specific machine. For the PC/XT/AT series of microcomputers, the ROM BIOS (Basic input/output system) and the operating system (typically PC-DOS, MS-DOS or UNIX) are proprietary and cannot be used without obtaining licensing rights. However, these proprietary rights are not at all limiting for the microcomputers being considered for CBI lab application because the ROM BIOS is included with the microcomputer and in most cases the operating system is also included. Additionally, the large installed base of MS-DOS and UNIX microcomputers has provided de facto operating system standards at relatively low cost. A multitasking operating system with powerful input/output capabilities and graphic screen paging are desirable for training and simulation applications. However, these operating system features are not essential for most CBI laboratory applications. Functions of this type can be built into applications programs when not available in the operating system without a severe efficiency penalty. Although it would appear to be beneficial for the Government to develop a simulation and training operating system for microcomputers, there is not an apparent immediate need to do so.

Languages

Programming languages are designed to serve as the communications and control interface between man and computers. A programmer uses a programming language to prepare a proceduralized list of tasks which can be interpreted by the computer to accomplish a computer function. Each language has a set of programming rules and a syntax structure which must be adhered to for effective program application and in certain cases, the rules and syntax are unique for specific applications.

Assembly language, which could be considered the original computer language, is closely related to machine architecture and is unique for each processor. It is also very elemental in the sense that data movement and processing operations must be implemented with instruction groups to accomplish useful functions. For this reason, it is considered a low level language, and it is used primarily by computer hardware and software designers. In the opposite sense, an authoring language would be designed for use by an instructional systems designer. The authoring language, to be effective, should not require the user to have an in-depth understanding of computer architecture. It should only be necessary that the user have an in-depth understanding of instructional system design.

Standard programming languages such as Pascal, Modula-2 or "C" are suitable for developing computer based instructional software, but their use would impair development efficiency. These languages have not been designed as instructional system development languages. Consequently, they do not provide programming tools in the form of functions and procedures peculiar to training. Although procedural adaptations can be developed to accomplish training functions, the adaptation would require redundant development for each independent development activity. For large scale application, this would become very inefficient.

If the use of CBI grows within the Navy operator and maintenance training community, it is recommended that a Government sponsored development program for a microcomputer based simulation and training language be considered. This language need not be any more than a library of training development unique "C" or ADA functions. In this way the language would be extensible and adaptive to the diverse needs of Navy training. A language of this type would enhance software transportability and improve the efficiency of training system modification and update.

The hardware and software technology needed to support simulation based operator and maintenance training are currently available. However, the development and implementation of this type of training on a large scale will require some changes in instructional system design philosophy. The software, hardware, and courseware designs should be treated as a single integrated system design using a design discipline similar to that used for weapon system design. In this way total system design integrity can be maintained. Without this systems design rigor, it is unlikely that all of the subsystem elements will fit together to effectively and efficiently satisfy Navy training requirements. In addition, the development risk would increase for separately developed subsystem elements. Hardware, software, and courseware require extensive integration for effective CBI development.

Many of the simulation and training hardware components to be integrated with software designs are currently available or can be developed without excessive cost or risk because of the extensive background the Navy has in this area of specialization. Standardized microcomputer operating systems are also available. If, in addition, effective simulation based authoring languages were to be developed by the Government for use by all contractors, training system prices should reduce to a level at which a wider application of this new technology should be possible.

CBI LABORATORY INSTRUCTIONAL GOALS AND METHODS

CBI laboratories can be used in achieving certain types of instructional goals in technical training. The most promising types of goals include (1) remediating students to meet required entry level knowledge, (2) job practice, (3) measuring student achievement, (4) refresher training, and (5) independent study of memory intensive skills.

Instructional methods used within CBI laboratories to accomplish these goals include (1) dialogue, (2) simulation, (3) games, (4) drill and practice and (5) tests. Within each of these methods there are variations in design that have special application to technical training.

These instructional goals and methods will be described in detail in this section of the report. Later sections will discuss the usefulness of various authoring languages and systems and will describe the types of equipment required to carry out these instructional methods.

INSTRUCTIONAL GOALS

CBI laboratories can be used in somewhat the same ways as workshops, equipment laboratories, testing centers, and language laboratories. Scheduled periods in the laboratory can be used to achieve special types of learning goals that cannot be conveniently learned in classrooms. The following description of Navy training goals and the ways in which the CBI laboratories can be used to achieve these goals will make clear the contributions CBI laboratories could make in Navy technical training.

Remediation for Entry Level Knowledge

The major cause of student failure in many training programs is that individual students lack the prerequisite knowledge for understanding new material being taught.

When individual students lack required mathematical skills, vocabulary or technical concepts, and the instructor assumes that all students have these prerequisites, then the students with inadequate background fall behind. Studies have shown that, of all the factors that influence the effectiveness of school learning, fifty percent of the variation in student performance in classroom instruction is due to variation in prerequisite knowledge. Overall school success rates would be much higher if students were tested for course and lesson prerequisites and then remediated, as necessary. Microcomputers in a CBI laboratory can be used to do the diagnostic testing and to deliver remedial instruction tailored to the individual student's need prior to the introduction of new material.

Job Practice

Some of the most successful training programs end with a laboratory assignment in which the students are presented with the tools of the trade to then perform specific job tasks on equipment in the laboratory. It is as if the students were on the job at assigned duty stations. Using technical training equipment (TTE) or simulators, the students operate or maintain equipment according to standard procedures. Prior to this final exercise, the students will have studied the theory of each operation, and should have practiced each operation separately on the TTE or a simulator.

In the typical Navy school this method does not work smoothly. A common problem is that students do not get enough hands-on time on the TTE. Even with the small class size of 6 or 8 students in the typical "C" school, scheduling adequate hands-on time is rarely possible. Purchasing the required number of units of operational equipment generally is too costly.

Computer simulations (microcomputers with graphic displays of a system) are being used to partially solve this problem. Electronic/electrical circuits, hydraulic, and pneumatic systems can be simulated with the microcomputer. With these simple simulations students mentally practice difficult procedures prior to their scheduled times on the TTE. The official technical manuals for the operations are used as job aids. Having practiced the operations on the computer, the hands-on time with the TTE is more effectively used, and higher skill levels, along with increased transfer and retention rates, are attained.

Microcomputer simulation, generally, does not replace the use of operational equipment in training. Instead, the use of computer simulation prepares the students to make more effective use of the limited time available for practice on operational equipment or expensive simulators.

Measuring Student Achievement

Measuring student success is an important part of Navy training, especially where the school's graduates must achieve a specified level of mastery before reporting to a duty station.

Manual testing is a labor intensive process for instructors. Instructors readily accept the use of computers to score tests, write student records, and prescribe and administer remediation to those individual students not meeting course standards.

Practical microcomputer-based testing now exists and, where used in the Navy, it is being well received by instructors and students. Testing programs are designed to measure academic achievements, i.e., skills in answering questions and in solving

problems that can be described and solved with words and numbers. And now, with the use of computer simulation, certain types of performance tests can be conducted.

Refresher Training

Students quickly forget certain types of information learned in Navy classrooms. Within a course, students forget what they are not required to recall and apply in follow-on lessons. It has been observed that in many courses there is no natural building up of course content from lesson to lesson; i.e., the lessons are independent blocks of instruction and can be presented in any order. It is under this condition that students do not retain what they have learned. Retention can be improved by forcing the recall and use of information, and distributing this practice at intervals throughout the remainder of the course.

With a CBI laboratory, it becomes practical to carry out this type of refresher training. Exercises can be carefully scheduled for students to practice recalling and applying easily forgotten information and to distribute this practice throughout the course in such a way that students will retain an ability to use this information when needed.

Independent Study of Memory Intensive Skills

Some skills cannot be effectively learned in a group setting. They must be acquired through extensive drill and practice by the individual student in addition to the usual classroom presentations. Typical tasks of this type include learning to touch type, learning Morse Code, and memorizing the characteristics of a wide range of foreign combatant vessels, including the radars they use, the weapons they carry, and the ranges of their detection systems and weapons. Other tasks which require drill and practice include learning to use standard computer software; such as word processors, data base managers, and spreadsheets; and learning to write programs in the various programming languages.

The CBI laboratory is best used for tasks which are difficult to learn, and require intensive memorization. CBI programs are effective because they not only present information, but also structure how the student is to learn the material. Many students cannot select efficient ways of learning difficult material and are helped when the program structures the learning process. The better programs provide for the application of new information in a range of problems which gradually increase in difficulty, with immediate evaluation of student responses, and remediation for wrong answers. However, this is too labor intensive for use with easily learned materials.

Other Observations

Any block of instruction can be accomplished in various ways. Too much of any learning approach, such as programmed instruction or lecture can become boring to students. And the lack of readiness, on the part of a bored student to continue to learn, will decrease the efficiency of the student's learning effort. The use of the CBI laboratory, as an adjunct to the traditional classroom, is one way of adding variety to the learning environment, and maintaining the student's readiness to learn.

The CBI laboratory can be used to carry out a broad range of instructional functions. Not only can the computer be used to generate a flow of practice materials, the content can be adjusted so that difficult material is practiced until learned and then repeated at intervals to ensure mastery. The use of the CBI laboratory to achieve many of these types of learning goals makes possible faster mastery of subject material, fewer failures, and better retention until the skills are used on-the-job. And for those jobs that require the use of a computer, practice at a computer work station is essential.

Table 1 summarizes types of instructional goals that can be satisfied in CBI laboratories. It also lists examples of training tasks that can be accomplished within each type of goal.

Table 1

Goals that can be Accomplished in CBI Laboratories

Types of Goals	Representative Tasks
Remediating Students to meet Required Entry Level Knowledge	Course Prerequisites o Mathematics o Typing o Technical vocabulary o Use of test equipment Lesson Prerequisites o Required information from previous lessons
Job Practice	Rehearsal for Practice with Operational Systems: o Equipment operating procedures o Casualty procedures o Fault isolation o Tactical decision making o Record keeping
Measuring Student Achievement	Measuring Academic Achievement o Lesson quizzes o Unit examinations o Final examinations Measuring Job Performance o Ability to perform equipment operating procedures o Ability to identify faults in electrical or electronic circuits o Ability to make tactical decisions under real time conditions.

Table 1 (Continued)

Types of Goals	Representative Tasks
Refresher Training	Scheduling and Administering Practice for those Concepts, Knowledge and Skills Quickly Forgotten by Students o Information not applied in subsequent lessons
Independent Study for Memory Intensive Skills	Complex Information that must be Memorized, Recalled and Applied Quickly and Accurately (especially where a keyed response is required) o Touch typing o Morse Code o Use of commands in computer programs such as Word Star or dBASE III o Use of computer programming languages to write software o Threat characteristics of foreign weapon systems

INSTRUCTIONAL METHODS

Five general instructional methods are being built into the programs to be used in CBI laboratories. These methods are (1) dialogue, (2) simulation, (3) games, (4) drill and practice and (5) tests. Each of these methods has significant variations. Also, some CBI programs use more than one method. In describing these methods, information will be presented on the major variations.

Dialogue

Conversation-like interactions between the student and the computer are called dialogue. The student uses the keyboard to carry on his side of the interchange. The computer evaluates the student's input, and responds with appropriate messages on the computer display. Two variations in this type of instruction are being used; tutorial dialogue, and inquiry dialogue.

With tutorial dialogue the computer presents information and asks questions, the student then responds by creating a short answer or selecting one of the programmed answers. The answer is entered into the computer via the keyboard, touch screen, or in some other way. The computer matches the student's response with stored correct answers or anticipated wrong answers. Appropriate conversation type messages are then displayed on the computer. If the computer program does not recognize the student's answer, the student is asked to clarify his response. Central to the success of this type of instruction is the writing of good code to cover a broad range of student responses. While true or false, matching, multiple choice, and fill-in-the-blank type questions are easier to program, limited free form language responses can be supported which will more closely approach human student-instructor dialogue.

Inquiry dialogue is similar to tutorial dialogue except that the student directly controls the flow of information from the computer. The student explores a problem by asking questions and collects the information which he feels will be helpful in solving the problem. This approach emulates an interview type format. Typical of this type of program are the exercises designed for doctors to use in practicing their diagnosing skills. The doctor is presented with a description of a patient, including apparent symptoms. The doctor then interviews the patient to get data on the patient's background, and more information on the symptoms. He can then order tests from a menu of available tests. After receiving the results of the tests, he can order more tests, or make a diagnosis and prescribe treatment. The patient's response to the treatment is also displayed. At the conclusion of the exercise the computer critiques the efficiency of the doctor in diagnosing and treating the patient.

Simulations

Microcomputer two dimensional (2D) simulations can be used by students to practice certain types of job-tasks and as tests for proper task performance. This form of simulation is especially useful when adequate practice time on the operational equipment or (3D) simulators cannot be scheduled, or when it is hazardous to practice on the operational equipment.

These microcomputer 2D simulations are typically graphic and alphanumeric models of a system displayed on a computer terminal. While 2D simulations do not provide the student with the tactile experience of throwing switches or turning dials, many types of Navy job-tasks can be recreated in this medium in such a way that mental practice can take place. The use of these computer simulations is usually considered as rehearsal for practice on the complex equipment. Often the objective with computer simulation is to prepare the student to make maximum use of available practice time on the operational system or the 3D simulator.

Appropriate training tasks for the Microcomputer

2D simulations include equipment operating procedures, malfunction diagnosis, repair procedures, and tactical decision making.

At least three major types of microcomputer simulations are appropriate for these purposes; dynamic system simulation, discrete task simulation, and scenario simulation. Each can accomplish special roles in technical training.

System simulations are dynamic mathematical or logical models of the behavior of a system. This behavior is generally depicted graphically and/or in alphanumeric form on a 2D computer display or on meters and other 3D devices. Because the behavior of the model is based on an interrelated set of equations, students are free to practice a range of operations and can find out what happens when wrong procedures are tried as well as correct procedures. This is sometimes called "free play" simulation. While this form of simulation represents the behavior of the real system under various conditions, it does not tutor the student in correct performance. Typical of this form of simulation are troubleshooting exercises in which the student is presented with a malfunctioning system. He can then check test points in any order, replace any part or module, and then rerun the system to see if it performs correctly.

Task simulations are designed to tutor students in the performance of specific operations, usually procedures. Simulations are used to instruct the student on procedure steps and will not let the student proceed until the steps are performed correctly. Unlike "free play" system simulations, the student must perform the operation in an approved manner.

Typical of this form is a procedure training exercise in which a student is tutored on the use of a checklist in performing an operator task on a piece of equipment. The student then practices the procedure on the simulation model. A computer graphics overview of the equipment is used as a menu for selecting close-up views. For example, the student can call up a close-up view of the area containing the power switch. He can then move the power switch to the on position. The light next to the power switch would then turn red indicating the power is on. If the student had taken any other action, it would have been ignored by the system, except for a message indicating an error. The student practices the entire procedure in this manner.

Scenario simulations are similar in structure to computer adventure games. The student is placed in an unfolding series of episodes or events, and each must be satisfied before moving to the next. For example, a student using a scenario simulation is assigned the role of Officer of the Deck aboard a vessel at night in the Hudson River leaving New York Harbor en route to the Atlantic Ocean. The student's task is to maneuver the vessel through the channel according to the rules of the road, and to avoid collisions with other vessels. A knowledge of navigation lights displayed by other vessels is necessary to understand the challenges that lay ahead. For instance, in one example, one set of lights may indicate that a vessel is crossing his bow and is unable to maneuver due to mechanical problems. The student analyzes his situation and takes some action. If he successfully completes the event, the next event in the scenario is presented.

Games

Computer-based training games serve useful roles in technical training. One main role is to make learning interesting, and to maintain a high level of student involvement. The development and use of interesting learning scenarios is a much needed quality in training programs, especially those programs which require the students to recall a large body of facts. These games generally make use of computer graphics, sound and attachments such as joysticks.

There are two broad types of training games, both useful in technical training. These are called strategy games and showdown games. Both types of training games are contests in which the players operate under formal rules with an objective of winning. In a computer game two or more players can compete, or the computer can play the role of the opposing player(s).

Strategy games are generally crude simulations of a competitive or conflict situation. As in the game of chess, each side must respond to the other side's moves. These responses are made by applying previously learned information.

The goal with this type of material is not to maximize realism, but to build a simplified model of a situation in which events happen rapidly. In this way the student has the maximum number of opportunities to recall and use the information being exercised with the game.

Showdown games are contests, such as 20 Questions, in which a student tries to do better than his previous performance, class norm or another student. As in the game of golf, each player performs independently, without interference from the other players. Each event is scored and the scores accumulated so that the student's overall score can be compared with other students' scores.

Drill and Practice

Students practice recalling difficult-to-remember information with drill and practice exercises. Examples of this type are exercises used by students to learn Morse code and weather symbols, or in solving mathematical problems. This type of exercise is generally used to increase the speed and accuracy of recalling information, especially when the information must be recalled over an extended period of time.

One type of information often practiced in these exercises involves stimulus-response pairs called "paired associates". Within these exercises the computer systematically manages the flow of items to the student according to schedules that have been found to be effective. Items are presented in a random order, and in small sets. The student receives immediate feedback on his performance during early training. As the student becomes more skilled, the feedback is gradually withdrawn until it matches that available on the job.

Tests

A variety of testing strategies can be implemented in the CBI laboratory. Academic and performance tests are the two types that will be most useful in Navy training. The primary reason for using computer-based testing is not in the area of test delivery, but in relieving instructors and administrators of labor intensive operations (i.e., test construction, test security, grading and remediation, maintaining student records, item analysis, and item banking). In the following paragraphs both academic and performance types of computer-based tests are discussed.

Academic tests are widely used in Navy technical training. Some of the more common types of tests include:

o Pretests - to determine if the student has the necessary prerequisite knowledge and to determine how much of the course content the students already know.

o Progress tests - to assess how well the students have mastered a block of instruction.

o Comprehensive within course tests - to assess how well the students have retained the content of a major phase of instruction.

o Final comprehensive test - to determine whether the objectives of a course have been attained and are still retained at the completion of a course.

Academic testing in the Navy is generally carried out with test items in the form of (1) multiple choice, (2) matching, (3) true or false, and (4) fill-in-the-blank. These forms of test items are easily created, banked, grouped into tests, administered, and analyzed in a computer-based environment. In the CBI laboratory the student can enter his answers directly into the computer or mark answer sheets to be read by optical scanners.

Performance tests measure how well individuals can do selected tasks. This form of testing is especially important in technical training where the purpose of instruction is to prepare students to perform jobs. In some training programs students are certified upon completion of the program as being able to perform specific tasks at specified skill levels. And in most courses, a Personnel Profile report summarizing a student's skills is forwarded to the student's duty station. In this context it is important for schools to test students' skill in performing job-tasks before certifying the students.

Two methods of performance testing are being used; process monitoring and results monitoring. Process monitoring requires that the instructor observe each step and then score this performance on a check sheet. It is an instructor-intensive type of testing. However, results monitoring requires less effort to score. The instructor merely checks the results of the student's performance after the test procedure has been completed, or at designated points within the procedure. Both types of performance testing can be carried out in the CBI laboratory. In the case of process monitoring the computer can be programmed to exercise the student in the procedure and then to verify that the student performed the procedure correctly, step by step, and within the required time limits. With this type of testing the instructor can observe each of the student's actions and ask questions as the student is carrying out the procedure.

With results monitoring type of testing, the computer is programmed to check the student's performance, but the instructor is not present to observe the student. The instructor gets a computer printout summarizing the student's work.

Summary

Table 2 summarizes instructional methods for use in CBI laboratories. Used as an adjunct to the Navy classroom, CBI laboratory exercises can be woven into the flow of course events in such a way that improved training effectiveness and greater transfer of training will result. In addition, the use of CBI laboratories should enable Navy instructors to help students acquire the skills needed to perform effectively in the fleet.

Table 2

Basic Instructional Methods Used in CBI Laboratories

Instructional Methods	Characteristics
Dialogue	Conversation-like Interchange Between Student and Computer.
Tutorial Dialogue	o Computer asks the questions. o Student answers. o Types of questions includes: - Multiple-choice. - Matching. - True or false - Fill-in-the-blank - Limited free form
Inquiry Dialogue	o Student asks the questions o Computer answers
Simulations	2D Microcomputer Models of Physical Systems Using Keyboard or Pointing Device Input and CRT for Alphanumeric and Graphic Display of System Dynamics 3D Microcomputer Models of Physical Systems Using Control System Inputs and 3D Device Outputs to Indicate System Dynamics
System Simulations	o Dynamic computer model of a system o Supports free play exercises o Typically does not tutor a student o Used for skill development and performance measurement
Task Simulation	o Computer model of a procedure o Tutors the student in performing a procedure o Restricts student to correct performance--not "free play" o Elements of Task Simulation can be embedded in System Simulation

Table 2 (Continued)

Instructional Methods	Characteristics
Scenario Simulation	o Places the student in a series of evolving episodes o Student must solve the problems in one episode before moving on
Games	Contests where Players Operate Under Formal Rules and the Objective is to Win. Exercises Students in Recall and Application of Information
Strategy Games	o Crude simulations of competitive or conflict situations o Opposing sides interact
Showdown Games	o Contests in which players perform independently o Scores are compared to determine winner
Drill and Practice	Practice Routines used to Increase the Speed and Accuracy in the Recall of Information and in the Application of Rules
Tests	Computer Routines that Administer Tests to Students and Perform Such other Operations as Supporting Test Construction, Test Security, Grading, Maintaining Student Records Item Analysis, and Item Banking
Academic Achievement Tests	o Pretests o Progress tests o Comprehensive within course tests o Final comprehensive tests
Performance Tests	o Process monitoring - Instructor observed and computer measured step by step performance of student o Results monitoring - Instructor checks a record of the process and the results of the operation - The student is not directly observed by the instructor

AUTHORING LANGUAGES AND AUTHORING SYSTEMS

A major decision to be made in designing courseware for a CBI laboratory is whether to use an authoring language or an authoring system. Both have features and limitations which will have an impact on the cost and effectiveness of the courseware being developed. The development of CBI lessons is typically the most expensive component of CBI laboratory development in that it is a labor intensive operation. This development expense also extends into the updating or revision of the courseware throughout the life cycle of a training program. It is important, therefore, to have a thorough understanding of the various types of authoring tools that influence courseware development costs.

AUTHORING LANGUAGES

Authoring languages provide flexibility in the development of courseware. In comparison to general purpose languages, authoring languages typically have special features which are designed to facilitate the production of courseware. While general purpose computer programming languages such as BASIC, PASCAL, C and ADA can be used for authoring, only skilled programmers are able to write the code for training programs. In addition, when courseware is developed with a general purpose language, it is often difficult to revise. In an attempt to simplify the courseware development process, authoring languages such as TUTOR (Control Data Corporation) PILOT (Multiple Vendors), TAL (TICCIT Authoring Language, Hazeltine Corporation), USE (Regency Systems, Inc.), and others have been developed. Overall, these authoring languages have procedures, functions, and utilities which are very useful in creating specialized training materials. Some general authoring language capabilities and limitations are listed below.

Authoring Language Features and Capabilities

1. Easier to learn than general purpose programming languages such as BASIC, PASCAL, or C.

2. Easier to use for developing new materials or revising existing materials than general purpose languages.

3. Provides for screen formatting, answer processing, data collection, and analysis.

4. Provides flexibility in programming; i.e., branching, use of videotape, videodisc, audio, high resolution graphics, animation, and simulation.

5. Supports both 2D and 3D sensing and control for simulation based lessons.

6. Incorporates preprogrammed graphic symbols--circles, boxes, lines, and special shapes.

7. Incorporates alternate character fonts for special effects, emphasis, or foreign language applications.

8. Incorporates graphic and text editors.

9. Provides multiple question formats; e.g., true or false, multiple choice, matching, and short answer.

10. Provides templates for performance-dependent branching which can be simple or complex.

11. Provides for giving answer-specific feedback for all question formats.

12. Can generate displays using multiple colors selected from a range of colors.

13. Can digitize visual images (photographs, training scenes).

14. Can provide a high degree of student interaction with lesson material through the use of general purpose or special purpose devices; e.g., keyboard, touch screen, light pen, mouse, digitizer pads, and 3D controls.

15. Can visually present detailed graphics with zoom, rotation, animation, and simulation capabilities.

16. Can track student performance.

17. Can maintain student records.

18. Can provide gaming oriented instruction; i.e., tactical situations between two opposing sides (student vs. student, student vs. instructor, or student vs. computer).

Limitations

1. Requires knowledge and skill in programming.

2. Requires knowledge and skill in instructional strategies.

3. Can sometimes constrain the author to relatively narrow methods of instruction.

4. Use is sometimes restricted because of single supplier proprietary rights.

5. Some require single source hardware components for authoring and student stations.

AUTHORING SYSTEMS

Authoring systems have been developed to make it easier to produce common types of courseware. They are typically structured systems. The courseware designer is given a limited number of authoring utilities with which certain instructional methods can be incorporated in courseware.

Options are often selected from a menu and the programming process is based upon the use of an instructional design syntax. An advantage, which can sometimes be a disadvantage, is that each authoring system must support specific types of lesson structures. Instructional strategies are often predetermined by the authoring system's developer and consequently are sometimes inflexible. Several authoring systems are presently available. Among them are the Regency Courseware Generator (Regency System Inc.), WISE (WICAT Systems), and ADAPT (Hazeltine Corp.). Some general features and capabilities of authoring systems are listed below.

Authoring System Features and Capabilities

1. Reduces the software development burden from the courseware designer and allows the designer to concentrate on lesson development.

2. Easier to learn than general purpose programming languages and authoring languages.

3. Provides a structured format for courseware design.

4. Standardizes lesson development.

5. Often uses a plain English syntax.

6. Requires only a minimum knowledge of computer programming.

7. Primarily used for academic types of lessons, but can be extended for simulation applications.

8. Provides flexibility in programming; i.e., branching, use of high resolution graphics, animation, simulation, videotape, videodisc, and audio.

9. Provides preprogrammed graphic symbols.

10. Incorporates alternate alphabet styles for special effects, emphasis, or foreign language applications.

11. Provides graphic and text editors.

12. Offers multiple question formats; e.g., true or false, multiple choice, matching, and short answer.

13. Provides answer-specific feedback for all question formats.

14. Can generate multicolor graphic displays.

15. Can display digitized high resolution imagery.

16. Can accept student responses from a number of input devices; e.g., keyboard, touch screen, light pen, mouse, etc.

17. Can incorporate material developed from a general purpose or an authoring language.

18. Reduces the development cost of standard types of courseware in comparison to courseware developed with authoring languages or general purpose languages.

Limitations

1. Usually limited to tutorial, drill and practice, and testing for applications; such as, the learning of basic skills (arithmetic, reading, and language arts).

2. Most require special hardware and software; i.e., specialized operating systems and student/instructor station hardware.

3. Typically, will not readily support procedural operations, system simulations, training games/tactical wargames, and performance testing.

4. Expensive to use because of special purpose hardware and software and also because of manufacturers' proprietary rights.

5. Uniqueness of special purpose systems results in high life cycle support costs.

Considerations for the Selection of an Authoring Language or System

In selecting an authoring system, it is important to consider the types of training objectives to be achieved. If the use of an authoring system will not satisfy the training objectives; e.g., simulation of operating or maintenance procedures, then an authoring language with simulation capabilities would be the proper choice. If either a language or a system will accomplish the training objective, then the choice

hinges on which will result in the lowest courseware cost. In addition, there is a personnel consideration. If the individuals who are developing the courseware do not have a good knowledge of programming and instructional design, an authoring system should be used. Even if personnel were given authoring language training, it has been found that such training does not rapidly make an individual a proficient programmer. For effective CBI development, personnel with both training and experience are required.

Individuals often learn to use an authoring language or system and then are expected to produce an instructional program in an unrealistically short period of time. Acquiring the knowledge and skills necessary to produce a good instructional program with an authoring language or system and then effectively applying these skills is not an easy task. However, if sufficient time is provided for the learning and development process, significant development efficiencies are possible which should more than offset training costs.

Authoring systems are normally used by teams consisting of learning specialists and subject matter experts with limited programming skills. The teams will generally be expanded to include professional computer programmers if an authoring language is used in place of an authoring system.

CBI LABORATORY SYSTEM CONFIGURATIONS

The CBI laboratory described in this report is envisioned as an interconnected complex of student station microcomputers and a management station microcomputer. The management station microcomputer can also serve as an authoring station for small laboratory applications. The typical laboratory would contain approximately 20 student stations, although this number could be reduced or increased to satisfy training requirements. If the training and curriculum development load is above average, it is recommended that a dedicated authoring capability be established in a separate area of the training facility. Normal laboratory power and air conditioning should be adequate for most laboratory applications with the exception of three dimensional (3D) simulation. If a number of high power consumption 3D devices are used in the laboratory, increased air conditioning and power distribution capacity might be required.

The CBI laboratory can be configured in many ways because of the many variations possible for each of the laboratory components. The options currently available for microcomputers, peripheral devices, operating systems, software, and networking components offer mix possibilities which go beyond practical design goals. For this reason the recommended configurations presented in this section are based on standardized components to the greatest extent possible.

SIMULATION

Two dimensional (2D) simulation utilizing a high resolution graphic display is recommended as the primary laboratory instructional design approach. The approach suggested for both 2D and 3D simulation is to use dynamic, real time, interactive delivery with a modeled play field. This "free play" type of continuous model has many advantages over the discrete event sequencing approach although development is often more labor intensive and implementations are more costly for laboratory training exercises. Simulation can take two general forms for this type of application. The first is typically used for procedure training wherein the accepted procedural steps are described by using flow charts or event lists. This type of training simulation can be mechanized in the form of procedure sequence tables and program event switching in response to student inputs.

The second form of simulation for training laboratory application is similar to that used for major device simulation. It is mathematical model based and capable of real time dynamic response to student inputs. Although the response is often artificial in appearance and typically presented in graphic form, it is very adaptive. Meter readings, status indications, digital readouts, and waveform displays can be presented for many different technical specialties. Both maintenance training and

operator training can be supported with this type of simulation and the range of training options are well beyond those offered with the discrete event simulation approach. The disadvantage of utilizing dynamic simulation for training is one of higher cost due to a restricted set of authoring tools. Because generic authoring systems are not available for this "free play" type of simulation, custom simulations are developed which are labor intensive and expensive. However, if effective authoring systems are developed for this form of simulation based training, development costs should not be significantly greater than discrete event simulation development costs.

For simulation described by mathematical models, it is not difficult to add 3D capabilities for input and output. The model is indifferent to the source of the input or the destination of the output. Consequently, 3D capabilities can be added for little more than the cost of microcomputer interface cards and 3D input, output devices. This technology has been available for many years, but it is only recently that the capabilities and costs of microcomputers have reached a point where cost-effective training applications are possible. Today, microcomputer based training technology is only used for critical need technical training applications. However, it is proposed that we are now at a juncture where wide scale application is possible at costs which are justifiable. Not only can 2D simulation be implemented cost-effectively, but 3D simulation based training is also a viable option. Simulation model outputs properly scaled and diverted to output ports can provide digital, or through conversion, analog outputs for drive to external devices. With a minimum number of standard interface boards it would be possible to include this capability as part of a simulation authoring module. There are a number of microcomputer based control systems currently in operation which use similar methods. It would be possible to adapt some of this hardware and software for training applications.

It has been demonstrated over the years that 2D and 3D simulation is a highly effective training medium. With some relatively inexpensive development, primarily centered around authoring system design, the power of simulation based training can be expanded to the Navy technical training community. Operational equipment is becoming very expensive and it is often cost prohibitive to use in Navy schools. Simulation based training, of the type proposed in this report can provide training alternatives which will give the student a sense of system dynamics during the training process. The CBI laboratory can be used for this type of application.

GRAPHICS

The effectiveness of the visual image has been demonstrated in many forms. Textbook illustrations, sound and slide presentations and more recently, video sequences are used extensively in all types of training. Today, the addition of low cost computer generated graphics adds a new dimension to this visual training approach. The most visible example of this dynamic graphics presentation approach is found in the many microcomputer games which are currently on the market. This technology, properly applied to training, can provide training capabilities never before possible. Not only can the student interact with the visual playfield in real time and develop skills needed on the job, but the cues and student responses can be scored, recorded, and analyzed in terms of skill level achievement.

Monochrome and color graphic display capabilites currently available, when used in conjunction with simulation based training sequences, offer a technical training potential which can not be fully assessed at this time. We only know that this type of training can be effectively delivered and managed at a lower cost than ever before. It appears that the obstacles to wide scale application have been essentially overcome. It is now a matter of designing instructional systems which can achieve the kinds of results desired. The computer graphic display is one of the key components used in effectively implementing this technology and it is also one of the most complex in terms of authoring graphics and delivering instruction. Some of the graphic display characteristics which are of importance in CBI laboratory design are addressed in this section.

The microcomputer displays used with the IBM and compatible PC/XT/AT architectures are of two types. The first is a monochrome character only, display which does not offer a graphics capability. This has the advantage of high resolution characters, formed by 9 x 14 dot cells, but the disadvantage of monochrome only and no graphics. The second is a color graphic display which offers 4 color graphics at 320 x 200 dot resolution, black and white graphics at 640 x 200 dot resolution and medium resolution characters formed by 8 x 8 dot cells. This is the preferred option for training because a graphic display capability is needed for most instructional applications. There are also graphic boards available by third party vendors which offer high resolution monochrome graphics, higher resolution color graphics and more color choices for each graphic mode. The use of these higher capability components might be the best choice for some training applications. However, only standard design PC/XT/AT interface boards will be discussed in this report in the interest of satisfying the standardization objective.

IBM has recently announced two new graphics systems referred to as enhanced graphics and professional graphics. These system offer color graphic dot resolutions, up to 640 x 480 with 256

display colors, selected from a palette of 4096 colors. With third party offerings, this resolution can be increased beyond 1000 x 1000 dot resolution with a wide range of color selection possibilities. However, this increased resolution and color option capability is expensive and probably not needed for most training applications. Experience to date has indicated that 320 x 200, 4 color and 640 x 200, black and white dot resolution will satisfy most training requirements. There are some applications such as dynamic instrument panel displays and map displays which might justify high resolution displays, but this option should be considered carefully before a decision is made. When high resolution is selected, the total laboratory configuration is affected. A single 1000 x 1000 dot resolution screen with 256 color possibilities for each picture element (PIXEL) will require a megabyte of memory. The screen refresh and update time required for a resolution of this magnitude, go beyond the capabilities of a dedicated graphics PC/XT/AT system. In addition, the storage requirements needed for hundreds or thousands of graphic images, go beyond reasonable storage capabilities for training laboratories. By comparison, a 640 x 200 dot resolution black and white, graphic image on a PC will only require 16K bytes to store a complete image. There are methods for overcoming the resolution and storage limitation, but none which appear feasible at this time.

Video disk and compact disk read only memory (CD-ROM) offer read only storage for hundreds of megabytes of graphic image data. However, this graphic information is of the read only variety and cannot be updated. This is a serious limitation for Navy training applications which require a 10 percent or more curriculum update each year. The best answer for laboratory applications, at this time, appears to be standard resolution displays with graphics drawing software. By storing drawings as codes for points, lines, and polygons, the storage requirements are dramatically reduced. The disadvantage of this approach is that the graphic screen must be redrawn each time it is displayed and for complex screens, this can take a few seconds. However, this disadvantage can be overcome with a multitasking system by drawing "next up" screens in background mode and instantaneously switching screens when needed.

The CD-ROM also offers some graphics potential not yet tapped for simulation purposes. Because of the small packaging requirement for the CD-ROM player and the small size of the changeable disk, it is well suited for student station use. Although the video disk and the CD-ROM both use similar technology, they are applied differently in that the discrete data on a video disk is converted to an analog television signal for use. The CD-ROM on the other hand uses the disk resident discrete data for digital data which can be communicated to a microcomputer as text or graphic information. With graphic processing software, this imagery could be used as a static simulation screen with a graphic overlay for imagery which must

respond dynamically. We are doing this today in a limited way by digitizing video disk information and overlaying dynamic graphics without the aid of an authoring system for simulation based graphic display. There are many potential training applications for this technology, and most of them are ready to implement. It appears that DOD needs to develop the authoring tools needed by instructional designers before the benefits of this emerging technology can be used effectively for training. This is recommended for consideration because microcomputers represent a fast moving technology in terms of new development.

LABORATORY CONFIGURATIONS

There are three recommended configurations for the CBI laboratory. These configurations will be referred to as:

a. Non networked
b. Low Speed Network
c. High Speed Network

The non networked laboratory configuration shown in figure 2 uses XT or AT microcomputers for student stations and an AT computer for an instructor/management/authoring station. The student stations have dual floppy drives and standard monochrome or color graphic displays. The student stations will also typically have a mouse or digitizer tablet for a pointing device. The courseware is provided on one floppy drive and the student work disk is used on the second drive to store student administrative, course sequencing and test results data. At the completion of an exercise, the student will bring the floppies to the instructor/management station, where the testing and performance measurement data is analyzed, and a report of results is prepared. This will also be accompanied by next assignment information, if follow-on modules are to be taken.

This relatively low-cost alternative suggests use of the instructor/management station, after laboratory hours, as an authoring station for developing new material and updating existing material. For all laboratory configurations, the effectiveness of the training materials developed will be highly dependent on the authoring and instructional management software provided. Although 3D simulation could be used with this laboratory alternative, it is not suggested unless hard disks are used for mass storage at the student stations.

This configuration has the benefit of easily adding student stations because networking is not used. It also has the benefit of student station portability. The student station can be placed in another laboratory, another training center or even aboard ship. This feature could prove valuable for afloat and remote station training. In terms of implementation effort and

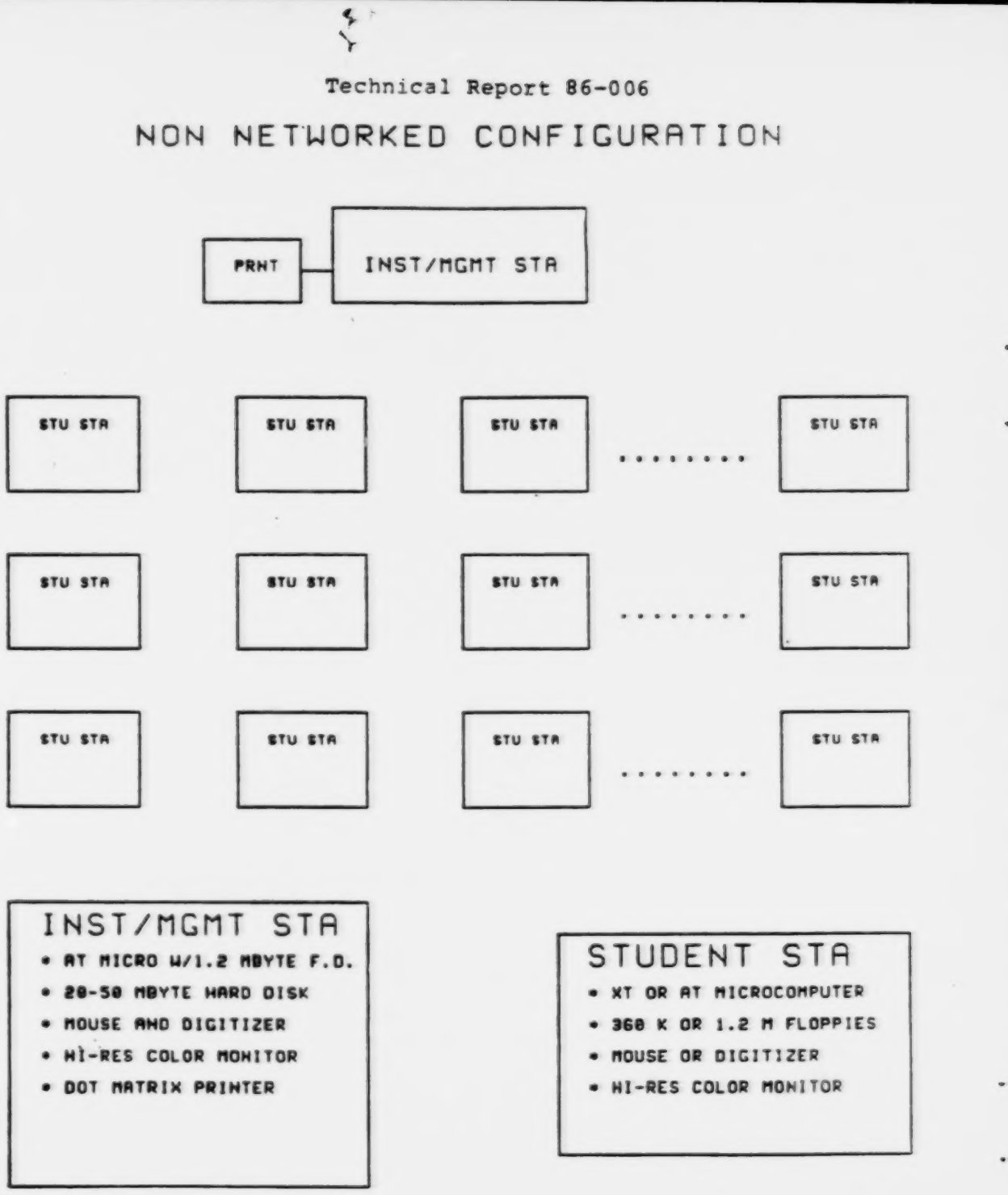


Figure 2. Instructional laboratory configuration No. 1

risk, this alternative would probably be the preferred option. However, the lack of mass storage and networking limits its use for many applications.

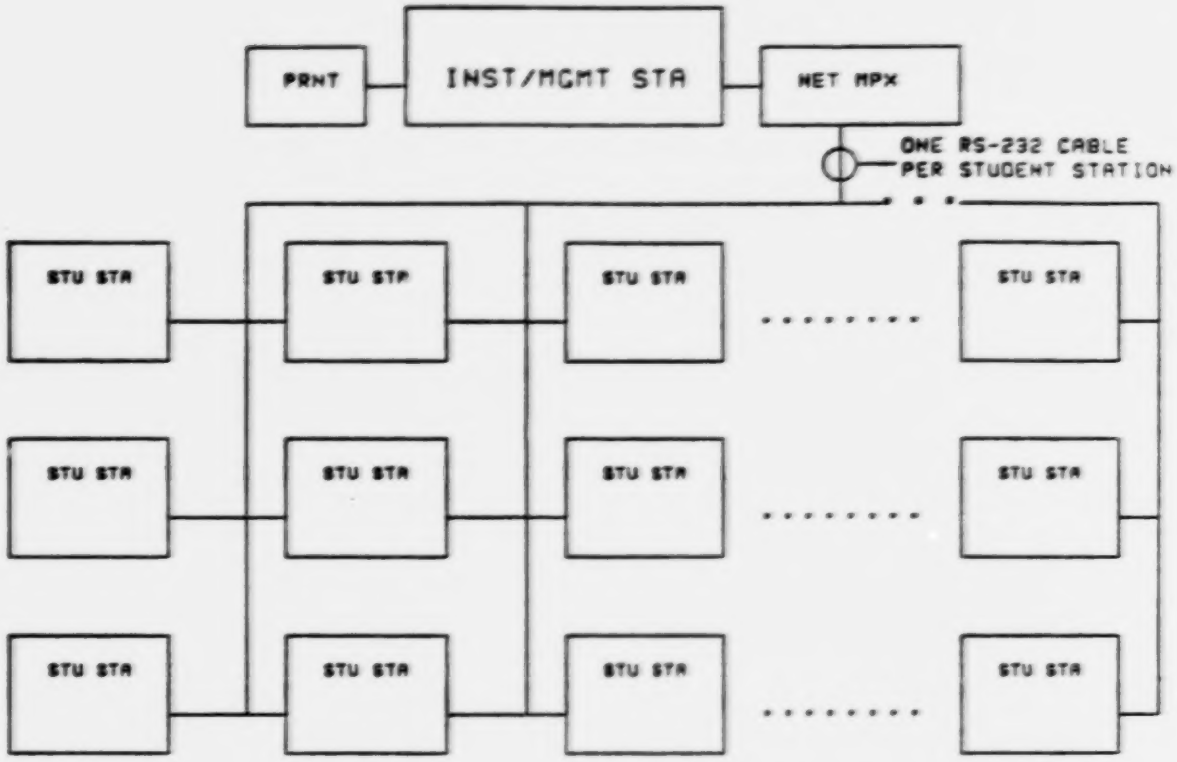
The low speed network configuration, figure 3, is essentially the same as the non networked configuration in terms of capabilities. Programs and student data disks are manually transported to the student station and returned to the instructor/management station at the end of a lesson. However, the instructor can monitor student progress during instructional sequences and student test data can be communicated over the network. Special student assignments can also be communicated from instructor to student, based on student progress. The network also allows for automated report generation for a group of students at the completion of an instructional or testing sequence. However, courseware, graphics data and simulation programs cannot be effectively communicated because of the network band width limitations and the low data rates for the RS-232 input/output channel.

The RS-232C interface proposed for this type of network is typically limited to data rates of about 2000 bytes per second. This is equivalent to communicating a full screen of text each second or a color graphics screen every 8 seconds. Since the polling and communications software at the instructor station will normally communicate with only one student station at a time, it is evident that 10 to 20 student stations could not communicate this type of information without excessive time delays. With a multitasking operating system and multiple input/output buffers, parallel student station communications would be possible, but the benefit to be gained is questionable. High volume data for graphic and simulation sequences would still be limited and the information needs for this option probably would not justify the added expenditure. This configuration would be the recommended option for low cost applications requiring laboratory network control.

The high speed network option shown in figure 4, costs more than the options discussed previously, but many benefits are added. For this option, an instructor/management station with the unix operating system or the yet to be released MS-DOS 5.0 operating system is suggested. It should also be supported with a fixed disk storage capacity of 50 to 100 Mbytes and a removable media tape drive for courseware loading and backup. The network suggested is the newly announced IBM microcomputer network, but any of the one to ten megabit per second networks would be suitable. Only one floppy per student station would be required because most information for courseware, simulation, graphics and student data would be communicated via the network. It is also likely that this single floppy would not be used for much beyond off-line diagnostics, initialization, and single station special assignments.

LOW SPEED (RS-232) NETWORK CONFIGURATION

Technical Report 86-006



INST/MGMT STA

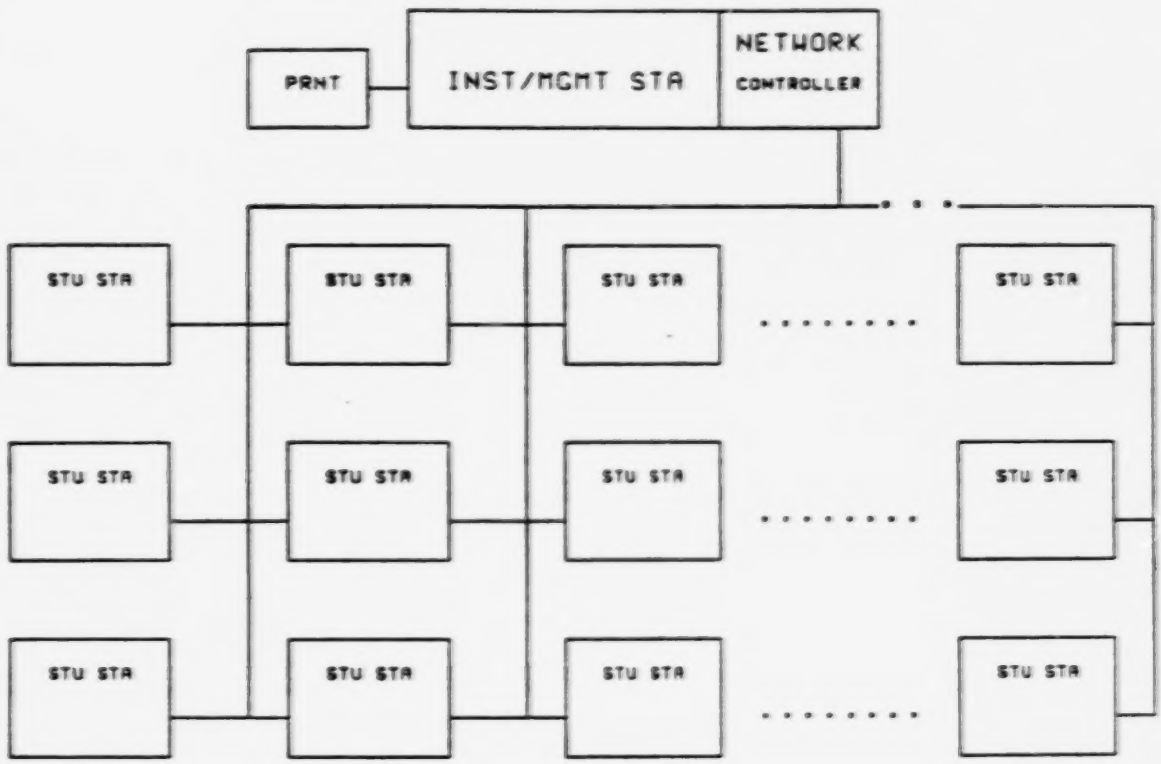
- AT MICRO W/1.2 MBYTE F.D.
- 20-30 MBYTE HARD DISK
- MOUSE AND DIGITIZER
- HI-RES COLOR MONITOR
- DOT MATRIX PRINTER
- RS-232 NET MULTIPLEXER

STUDENT STA

- XT OR AT MICROCOMPUTER
- 360 K OR 1.2 M FLOPPIES
- MOUSE OR DIGITIZER
- HI-RES COLOR MONITOR
- RS-232 INT WITH NET S/W

Figure 3. Instructional laboratory configuration No. 2

HIGH SPEED NETWORK CONFIGURATION



INST/MGMT STA

- AT MICRO W/1.2 MBYTE F.D.
- 20-30 MBYTE HARD DISK
- MOUSE AND DIGITIZER
- HI-RES COLOR MONITOR
- DOT MATRIX PRINTER
- NETWORK CONTROLLER

STUDENT STA

- XT OR AT MICROCOMPUTER
- 360 KB OR 1.2 MB FLOPPY
- MOUSE OR DIGITIZER
- HI-RES COLOR MONITOR
- NETWORK INTERFACE

Figure 4. Instructional laboratory configuration No. 3

The advantages to be gained by a configuration of this type go beyond laboratory instruction. With a network of this type it is also possible to locate an authoring station remotely and communicate new materials and revised materials via network. In a like manner, the network can be used to communicate administrative information to a school's command administrative office.

This option is also the one suggested for intensive 2D or 3D simulation which will often require high data rate access to simulation programs and data bases. Although any of the instruction delivered with this configuration could be duplicated with the low cost non networked option, it would require the addition of a hard disk at each student station. This additional cost added to the disadvantage of manually transporting floppies or tapes between stations would appear to offset the advantages. It is expected that this configuration will become the most popular and possibly the most cost-effective because of the marginal benefit to cost ratio for the added capability.

The laboratory configurations suggested in this report are based on some limited experience with CBI applications in the Navy. None have achieved the training effectiveness potential which is considered possible today. However, it is expected that significant training effectiveness improvements and cost reductions will occur during the next decade. It is hoped that the information in this report can be used in support of this movement toward greater instructional efficiency in the Navy.

DISTRIBUTION LIST

Navy

ASSTSECNAV (MRA)
CNO (OP-11, OP-01B7, OP-11H3, OP-12, OP-29, OP-39, OP-59, NOP-0987H, NOP-914D, NOP-987)
CNR (170, 442 (3 copies))
CNET (01, 00A, N-1, N-2, N-3, L02)
CNATRA (Library (2 copies))
COMNAVAIRSYSCOM (APC 205-0, APC 205-1)
CNTECHTRA (N-6 (2 copies))
CO FLEMINEWARTRACEN
CO FLETRACEN, Mayport
CO FLETRACEN, San Diego
CO FLETRACEN, Norfolk
COMFLETRAGRU, Pearl Harbor
COMFLETRAGRU, San Diego
CO NATTC
AEGIS TRACEN (CISCO)
CO NAVPERSRANDCEN (Library (3 copies))
COMNAVRESFOR (Code 02)
COMNAVAIRRESFOR
COMNAVSURFRESFOR
CO NAVSUBTRACENPAC (2 copies)
CO SERVSCOLCOM, Orlando
CO SERVSCOLCOM, San Diego
CO SERVSCOLCOM, Great Lakes
CO FLETRACEN San Diego
COMTRALANT (00, Ed. Adv.)
COMPTRAPAC (2 copies)
CO TRITRAFAC, Bangor (2 copies)
CO FCTCLANT
CO FCTCPAC
CO FLEASWTRACENLANT
CO FLEASWTRACENPAC
CO HSL 31 (Trng. Dept.)
CO NAVAEROMEDRSCHLAB (Aviation Psy. Div.)
CO NAVAEROSPMEDINST (Code 11, Code 13)
CO NAVAVSCOLSCOM (Code 0212)
SUP NAVPGSCOL (Code 2124, Code 32)
COMNAVSHIPYD, Charleston (180 MST)
COMNAVSHIPYD, Pearl Harbor (Code 302.3)
CO SHIPYDTRSUPPCENLANT (Code 1400)
CO NAVSPSERVADMINACT (Trng. Sup. Unit)

DISTRIBUTION LIST (Continued)

CO NAVSWC (Code E411)
CO NAVTECHTRACEN, Corry (Code 101B, Code 3330,
Crypto. Trng. Dept.)
CO NAVTECHTRACEN, Meridian
CO NAVTECHTRACEN, TI (Tech. Lib.)
USNA (Behavioral Science Dept.)
CO NMITC
CO NAVTRASYSCEN (Code 001, 002, 09, 1, 4L, 4P, 7B, TIC,
Code 10 (3 copies) Code 123 (5 copies))

Marine Corps

CMC (OT)
CG MCDEC (IMSEDCTR, Ed. Adv.)
MARCORCOMMELESCOL (SCOLOPNSDEPT)
DIRMCI

Army

ARI (Library (2 copies), PERI-IC, RESINST, Ref. Serv.)
ARI, Fort Knox (PERI-IK)
CG, TRADOC (Tech. Lib.)
CMDT USAIPRM (ATSG-DT-R)

Air Force

HQ ATC/TTXA
USAFOMC/OMT
AFMTC/XR
AFOSR/NL
4235 STS/DOTE
HQ TAC/DOOS

Coast Guard

COMDT COGARD (G-P-1/2/42, GRT/54)

DISTRIBUTION LIST (Continued)

Other

CNA
DLA
USD/R&E
IDA

Information Exchanges

DTIC (12 copies)
ERIC (2 copies)
DLSIE
APA